

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА**

ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

КАФЕДРА ІНТЕЛЕКТУЛЬНИХ ТЕХНОЛОГІЙ

Лабораторна робота №1

з дисципліни «Непроцедурне програмування»

Тема: Мова Haskell. Робота з інтерпретатором ghci

Варіант 6

Виконала студентка групи КН-31
Мензатюк Олександра Петрівна

Перевірв:
Миколайчук Роман Антонович

Київ – 2021

Мета: Ознайомитись з основними типами мови. Ознайомитись зі структурою та функціями Glasgow Haskell Compiler. Набути навичок роботи з інтерпретатором ghci та визначення найпростіших функцій.

Завдання 1

Наведіть приклади виразів вказаного типу. Кожен список має містити кілька елементів. Перегляньте тип прикладів, як їх визначає ghci. Прокоментуйте.

`[(Integer, (Char,[Bool]))]`

Код:

```
a ::[(Integer, (Char, [Bool]))]
a = [(13, ('a',[True, 1==1, False>True])), (-1, ('0',[True, True, False>True]))]

b ::[(Integer, (Char, [Bool]))]
b = [(111, ('\t',[isDigit 'a', isDigit '1', 'a'=='b'))), (0, ('\n',[False /= False,
(&&) True True, (||) False True]))]
```

Протокол тестування:

```
*Main> a
[(13,('a',[True,True,False])),(-1,('0',[True,True,False]))]
*Main> :type a
a :: [(Integer, (Char, [Bool]))]
*Main> b
[(111,('\t',[False,True,False])),(0,('\n',[False,True,True]))]
*Main> :type b
b :: [(Integer, (Char, [Bool]))]
```

Завдання 2

Визначте два варіанти вказаних далі функцій. Перший варіант – з одним аргументом-кортежем, другий – без використання кортежів чи списків.

Функція визначає, чи одне коло повністю міститься в іншому. Кожне коло задається координатами центра та радіусом.

Код:

```
function1 :: (Double, Double, Double, Double, Double, Double) -> Bool
function1 (x1, y1, r1, x2, y2, r2) = sqrt ((x2-x1)^2 + (y2-y1)^2) <= abs(r1-r2)

function2 :: Double -> Double -> Double-> Double -> Double -> Double -> Bool
function2 x1 y1 r1 x2 y2 r2 = sqrt ((x2-x1)^2 + (y2-y1)^2) <= abs(r1-r2)
```

Протокол тестування:

```
*Main> function1 (0, 0, 1, 0, 0, 2)
True
*Main> function1 (0, 0, 1, 1, 1, 2)
False
*Main> function1 (0, 1, 1, 1, 1, 2)
True
*Main> function1 (-1, -1, 1, 10, 10, 2)
False
*Main> function1 (0, 0, 2, 0, 0, 1)
True
*Main> function2 0 0 1 0 0 2
True
*Main> function2 0 0 1 1 1 2
False
*Main> function2 0 1 1 1 1 2
True
*Main> function2 (-1) (-1) 1 10 10 2
False
*Main> function2 0 0 2 0 0 1
True
```

Висновок

У ході виконання лабораторної роботи №1 було наведено приклади виразів різного типу, що дозволило ознайомитися і зрозуміти типізацію в мові Haskell. Також було визначено і протестовано функції, на вхід однієї з яких подається кортеж, а на вхід другої – звичайний набір аргументів.